WHAT IS CLAIMED IS:

1. A method of assigning virtual memory to physical memory in a data processing system, comprising the steps of:

allocating a set of physical memory pages of the data processing system for a new virtual superpage mapping;

instructing a memory controller of the data processing system to move a plurality of virtual memory pages corresponding to an old page mapping to the set of physical memory pages corresponding to the new virtual superpage mapping; and

accessing at least one of the virtual memory pages using the new virtual superpage mapping while the memory controller is copying old physical memory pages to new physical memory pages.

2. The method of Claim 1 wherein said allocating step allocates a contiguous set of physical memory pages.

3. The method of Claim 1 wherein said accessing step includes the step of directing a read operation for an address of the new page mapping which is currently being copied to a corresponding address of an old page mapping.

4. The method of Claim 1 wherein said accessing step includes the step of directing a write operation for an address of the new page mapping which is currently being copied to both the address of the new page mapping and a corresponding address of an old page mapping.

5. The method of Claim 1 wherein said accessing step includes the step of directing a write operation for an address of the new page mapping which has not yet been copied to a corresponding address of an old page mapping.

6. The method of Claim 1, further comprising the step of updating an entry in a cache memory of the data processing system which corresponds to a memory location in the virtual memory page, by modifying an address tag of the cache entry according to the new page mapping.

7. A memory controller comprising:

an input for receiving page remapping instructions;

a mapping table which temporarily stores entries of old page addresses and corresponding new page addresses associated with the page remapping instructions; and

a memory access device which directs the copying of memory pages from the old page addresses to the new page addresses and releases the entries in said mapping table as copying for each entry is completed.

8. The memory controller of Claim 7 wherein said mapping table has 32 slots for receiving corresponding pairs of the old page addresses and new page addresses.

9. The memory controller of Claim 7 wherein said memory access device directs a read operation for a new page address which is currently being copied to a corresponding old page address.

10. The memory controller of Claim 7 wherein said memory access device directs a write operation for a new page address which is currently being copied to both the new page address and a corresponding old page address.

11. The memory controller of Claim 7 wherein said memory access device directs a write operation for a new page address which has not yet been copied to a corresponding old page address.

12. The memory controller of Claim 7 wherein said memory access device includes a state engine which sequentially reads the paired old and new pages addresses in said mapping table.

13. The memory controller of Claim 12 wherein said memory access device further includes a direct memory access (DMA) engine controlled by said state machine which carries out actual copying of the memory pages.

14. A computer system comprising:

a processing unit;

an interconnect bus connected to said processing unit;

a memory array; and

a memory controller connected to said interconnect bus and said memory array, wherein said memory controller copies memory pages from old page addresses to new page addresses while said processing unit carries out program instructions using the new page addresses.

15. The computer system of Claim 14 wherein:

said processing unit includes a processor core having a translation lookaside buffer (TLB) whose entries keep track of current virtual-to-physical memory address assignments; and

said TLB entries are updated for the new page addresses prior to completion of copying of the memory pages by the memory controller.

16. The computer system of Claim 14 wherein:

said processing unit has a processor core and an associated cache; and

said cache modifies an address tag of a cache entry which corresponds to a memory location in the new page addresses.

17. The computer system of Claim 16 wherein said cache modifies the address tag of the cache entry in response to a determination that the cache entry contains a valid value which is not present elsewhere in the system.

18. The computer system of Claim 16 wherein said cache further relocates the cache entry based on a changed congruence class for the modified address tag.

19. The computer system of Claim 14 wherein said memory controller includes:

a mapping table which temporarily stores entries of old page addresses and corresponding new page addresses; and

a memory access device which directs the copying of the memory pages from the old page addresses to the new page addresses and releases the entries in said mapping table as copying for each entry is completed.

20. The computer system of Claim 14 wherein said processing unit, said interconnect bus, said memory array and said memory controller are all part of a first processing cluster, and further comprising a network interface which allows said first processing cluster to communicate with a second processing cluster, said memory controller having at least one pointer for a new page address which maps to a physical memory location in said second processing cluster.

21. A cache memory for a processing unit, comprising:

a data array which stores values associated with respective locations in system memory;

a tag array which stores address tags corresponding to the values in said data array;

a cache controller which receives cache instructions and accesses said data array and said tag array to carry out read and write operations; and

a state machine which modifies an address tag for a cache entry based on a new memory mapping for an associated memory location of the cache entry.

22. The cache memory of Claim 21 wherein said state machine modifies the address tag in response to a determination that the cache entry has a dirty coherency state.

23. The cache memory of Claim 21 wherein said cache controller relocates the cache entry based on a changed congruence class for the modified address tag.